UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/668,610	09/22/2000	Carl M. Ellison	042390.P8104X	2283

8791 7590 06/28/2005

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
ABRISHAMKAR, KAVEH	
ART UNIT	**PAPER NUMBER**
2131	

DATE MAILED: 06/28/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/668,610	ELLISON ET AL.	
	Examiner	Art Unit	
	Kaveh Abrishamkar	2131	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 April 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-8,10-20,22-32 and 34-48* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2,4-8,10,11,13,14,16-20,22,23,25,26,28-32,34,35,37,38,40-44,46 and 47* is/are rejected.

7)☒ Claim(s) *3,12,15,24,27,36,39 and 48* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *01/18/2005*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is in response to the Request for Continued Examination (RCE) filed on April 4, 2005. Claims 9,21,33, and 45 were cancelled per the amendment received on July 23, 2004. Per the RCE response, claims 1-3,11,13-15,23,25-27, 35, and 37-39 are currently amended. Claims 1-8, 10-20, 22-32, and 34-48 remain pending in the application.

Information Disclosure Statement

2. An initialed and dated copy of Applicant's IDS form 1449, received January 18, 2005, is attached to this Office action. The reference list accompanying the IDS form 1449, received on December 16, 2003, is missing, and Examiner respectfully requests a copy of the reference list with the next communication so that it can be considered.

Allowable Subject Matter

3. Claim 3,12, 15, 24, 27, 36, 39, and 48 objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-2, 4-6, 10-11, 13-18, 22-23, 25-26, 28-30, 34-35, 37-39, 40-42, and 46-47 rejected under 35 U.S.C. 102(e) as being anticipated by England et al. (U.S. Patent 6,327,652).

Regarding claim 1, England discloses:

An apparatus comprising:

a key generator to generate an operating system nub key (OSNK) unique to an operating system (OS) nub, the OS nub being part of an operating system to run on a platform comprising a processor capable of operating in an isolated execution mode in a ring 0 operating mode, wherein the processor also supports one or more higher ring operation modes, as well as a normal execution mode in at least the ring 0 operating mode; (Figure 8 item 801, column 7 line 45-61, column 17 lines 1-15); and

a usage protector coupled to the key generator to protect usage of a subset of a software environment using the OSNK (column 17 line 1 – column 18 line 13);

wherein the usage protector performs at least one operation selected from the group consisting of:

encrypting a value while operating in isolated execution mode (column 15 line 61 – column 16 line 67); and

decrypting an encrypted value while operating in isolated execution mode (column 15 line 61 – column 16 line 67).

England discloses an apparatus that uses an Operating System (OS) key to secure access to an operating system operating in a secure mode. Furthermore, England describes that "an unrelated operating system cannot gain access to the encrypted data" (column 17 lines 54-58) because of the requirement of the OS key.

Regarding claim 13, England discloses:

A method comprising:

generating an operating system nub key (OSNK) unique to an operating system (OS) nub, the OS nub being part of an operating system to run in a software environment on a platform comprising a processor capable of operating in an isolated execution mode in a ring 0 operating mode, wherein the processor also supports one or more higher ring operating modes, as well as a normal execution mode in at least the ring 0 operating mode; (Figure 8 item 801, column 7 line 45-61, column 17 lines 1-15); and

protecting usage of a subset of the software environment using the OSNK (column 17 line 1 – column 18 line 13);

wherein the operation of protecting usage of a subset of a software environment comprises at least one operation selected from the group consisting of:

encrypting a value while operating in isolated execution mode (column 15 line 61 – column 16 line 67); and

decrypting an encrypted value while operating in isolated execution mode (column 15 line 61 – column 16 line 67).

England discloses an apparatus that uses an Operating System (OS) key to secure access to an operating system operating in a secure mode. Furthermore, England describes that "an unrelated operating system cannot gain access to the encrypted data" (column 17 lines 54-58) because of the requirement of the OS key.

Regarding claim 25, England discloses:

A computer program comprising:

a computer usable medium having computer program code embodied therein, the computer program product having:

computer readable program code to generate an operating system nub key (OSNK) unique to an operating system (OS) nub, the OS nub being part of an operating system to run in a software environment on a platform comprising a processor capable of operating in an isolated execution mode in a ring 0 operating mode, wherein the

processor also supports one or more higher ring operating modes, as well as a normal execution mode in at least the ring 0 operating mode; (Figure 8 item 801, column 7 line 45-61, column 17 lines 1-15); and

computer readable program code for protecting usage of a subset of the software environment using the OSNK (column 17 line 1 – column 18 line 13);

wherein the operation of protecting usage of a subset of the software environment comprises at least one operation selected from the group consisting of:

encrypting a value while operating in isolated execution mode (column 15 line 61 – column 16 line 67); and

decrypting an encrypted value while operating in isolated execution mode (column 15 line 61 – column 16 line 67).

England discloses an apparatus that uses an Operating System (OS) key to secure access to an operating system operating in a secure mode. Furthermore, England describes that "an unrelated operating system cannot gain access to the encrypted data" (column 17 lines 54-58) because of the requirement of the OS key.

Regarding claim 37, England discloses:

A system to provide a secure platform, the system comprising:

a processor capable of operating in an isolated execution mode in a ring 0 operating mode, wherein the processor also supports one or more higher ring operating

modes, as well as a normal execution mode in at least the ring 0 operating mode; (Fig 1B item 160, column 7 line 44-50);

storage response to the processor, the storage storing at least a subset of a software environment to run on the system (Fig 1B item 184, column 17 line 1 – column 18 line 13);

an operating system (OS) nub (Figure 8 item 801, column 7 line 45-61, column 17 lines 1-15);

a key generator to generate a operating system nub key (OSNK) unique to the operating system (OS) nub (Figure 8 item 801, column 7 line 45-61, column 17 lines 1-15); and

a usage protector coupled to the key generator to protect usage of a subset of the software environment using the OSNK (column 17 line 1 – column 18 line 13);

wherein the operation of protecting usage of a subset of the software environment comprises at least one operation selected from the group consisting of:

encrypting a value while operating in isolated execution mode (column 15 line 61 – column 16 line 67); and

decrypting an encrypted value while operating in isolated execution mode (column 15 line 61 – column 16 line 67).

England discloses an apparatus that uses an Operating System (OS) key to secure access to an operating system operating in a secure mode. Furthermore, England

describes that "an unrelated operating system cannot gain access to the encrypted data" (column 17 lines 54-58) because of the requirement of the OS key.

Claim 2 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the key generator comprises:

a combiner to combine an identification of the OS nub and a master binding key (BK0) of the platform, the combined identification and the BK0 corresponding to the OSNK (column 12 line 53-65).

England discloses providing a one-way hashing function of the loaded components of the secure OS and then signing the hash with a private key corresponding to the operating system components.

Claim 4 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the usage protector comprises:

an encryptor to encrypt the subset of the software environment using the OSNK, the encrypted subset being stored in a storage (column 7 lines 44- 50, column 13 lines 10-59); and

a decryptor to decrypt the encrypted subset using the OSNK, the encrypted subset being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59).

Claim 5 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim wherein the usage protector comprises:

an encryptor to encrypt a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

a decryptor to decrypt the encrypted first hash value using the OSNK, the encrypted first hash value being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

a comparator to compare the decrypted first hash value to a second hash value to generate a compared result, the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 6 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the usage protector comprises:

a first encryptor to encrypt a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

a second encryptor to encrypt a second hash value using the OSNK (column 7 lines 44-50, column 13 lines 10-59); and

a comparator to compare the encrypted second hash value to the encrypted first hash value to generate a compared result, the encrypted first hash value being retrieved from the storage, the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 10 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the software environment comprises an operating system selected from the group consisting of a Windows operating system, a Windows 95 operating system, a Windows 98 operating system, a Windows NT operating system, and a Windows 2000 operating system (column 21 lines 25-29).

Claim 11 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the subset of the software environment comprises a registry of an operating system (column 13 lines 37 – 53).

Claim 14 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein generating the OSNK comprises:

combining an identification of the OS nub and a master binding key (BK0) of the platform, the combined identification and the BK0 corresponding to the OSNK (column 12 line 53-65).

Claim 16 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein protecting usage comprises:

encrypting the subset of the software environment using the OSNK (column 7 lines 44- 50, column 13 lines 10-59);

storing the encrypted subset in a storage (column 7 lines 44- 50, column 13 lines 10-59); and

decrypting the encrypted subset from the storage using the OSNK (column 7 lines 44-50, column 13 lines 10-59).

Claim 17 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein protecting usage comprises:

encrypting a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

decrypting the encrypted first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

comparing the decrypted first hash value to a second hash value to generate a compared result, the decrypted first hash value being retrieved from the storage, the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 18 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein protecting usage comprises:

encrypting a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

encrypting a second hash value using the OSNK (column 7 lines 44-50, column 13 lines 10-59); and

comparing the encrypted first hash value to the encrypted second hash value to generate a compared result, the encrypted first hash value being retrieved from the storage, the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 22 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein the software environment comprises an operating system selected from the group consisting of a Windows operating system, a Windows 95 operating system, a Windows 98 operating system, a Windows NT operating system, and a Windows 2000 operating system (column 21 lines 25-29).

Claim 23 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein the subset of the software environment is a registry of the operating system (column 13 lines 37 – 53).

Claim 26 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

the computer program of claim 25 wherein the computer readable program code for generating the OSNK comprises:

computer readable program code for combining an identification of the OS nub and a master binding key (BK0) of the platform, the combined identification and the BK0 corresponding to the OSNK (column 12 line 53-65).

Claim 28 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

The computer program product of claim 25 wherein the computer readable program code for protecting usage comprises:

computer readable program code for encrypting the subset of the software environment using the OSNK (column 7 lines 44- 50, column 13 lines 10-59);

computer readable program code for storing the encrypted subset (column 7 lines 44- 50, column 13 lines 10-59); and

computer readable program code for decrypting the encrypted subset from the storage using the OSNK (column 7 lines 44-50, column 13 lines 10-59).

Claim 29 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

The computer program product of claim 25 wherein the computer readable program code for protecting usage comprises:

computer readable program code for encrypting a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in storage (column 7 lines 44-50, column 13 lines 10-59);

computer readable program code for decrypting the encrypted first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

computer readable program code for comparing the decrypted first hash value to a second hash value to generate a compared result, the decrypted first hash value being retrieved from the storage, the compared result indicating whether the subset of

the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 30 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

The computer program product of claim 25 wherein the computer readable program code for protecting usage comprises:

computer readable program code for encrypting a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in storage (column 7 lines 44-50, column 13 lines 10-59);

computer readable program code for encrypting a second hash value using the OSNK (column 7 lines 44-50, column 13 lines 10-59); and

computer readable program code for comparing the encrypted first hash value to the encrypted second hash value to generate a compared result, the encrypted first hash value being retrieved from the storage , the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 34 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

The computer program product of claim 25 wherein the software environment comprises an operating system selected from the group consisting of a Windows

operating system, a Windows 95 operating system, a Windows 98 operating system, a Windows NT operating system, and a Windows 2000 operating system (column 21 lines 25-29).

Claim 35 is rejected as applied above in rejecting claim 25. Furthermore England discloses:

The computer program product of claim 25 wherein the subset of the software environment comprises a registry of an operating system (column 13 lines 37 – 53).

Claim 38 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the key generator comprises:

a combiner to combine an identification of the OS nub and a master binding key (BK0) of the system, the combined identification and BK0 corresponding to the OSNK (column 12 line 53-65).

Claim 40 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the usage protector comprises:

an encryptor to encrypt the subset of the software environment using the OSNK, the encrypted subset being stored in a storage (column 7 lines 44- 50, column 13 lines 10-59); and

a decryptor to decrypt the encrypted subset using the OSNK, the encrypted subset being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59).

Claim 41 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the usage protector comprises:

an encryptor to encrypt a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

a decryptor to decrypt the encrypted first hash value using the OSNK, the encrypted first hash value being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

a comparator to compare the decrypted first hash value to a second hash value to generate a compared result, the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 42 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the usage protector comprises:

a first encryptor to encrypt a first hash value of the subset of the software environment using the OSNK, the encrypted first hash value being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

a second encryptor to encrypt a second hash value using the OSNK (column 7 lines 44-50, column 13 lines 10-59); and

a comparator to compare the encrypted second hash value to the encrypted first hash value to generate a compared result, the encrypted first hash value being retrieved from the storage, the compared result indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

Claim 46 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the software environment comprises an operating system selected from the group consisting of a Windows operating system, a Windows 95 operating system, a Windows 98 operating system, a Windows NT operating system, and a Windows 2000 operating system (column 21 lines 25-29).

Claim 47 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the subset of the software environment comprises a registry of an operating system (column 13 lines 37 – 53).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 7-8, 19-20, 31-32, and 43-44 are rejected under 35 U.S.C. 103(a) as being unpatentable over England et al. (U.S. Patent 6,327,652).

Claim 7 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the usage protector comprises:

a decryptor to decrypt a protected private key to generate a private key using the OSNK (column 7 lines 44-50, column 13 lines 10-59);

a signature generator coupled to the decryptor to generate a signature of the subset of the software environment using the private key, the signature being stored in a storage (column 7 lines 44-50, column 13 lines 10-59); and

a signature verifier to verify the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the subset has been modified (column 7 lines 44-50, column 13 lines 10-59).

England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." It is obvious that the

capability exists in the apparatus of England to decrypt a protected private key. Also, it is described that the processor has the capability to generate signatures, and the verification procedure for a signature is analogous to the comparator of the one-way hash functions.

Claim 8 is rejected as applied above in rejecting claim 1. Furthermore, England discloses:

The apparatus of claim 1 wherein the usage protector comprises:

a manifest generator to generate a manifest of the subset of the software environment, the manifest describing the subset of the software environment, the manifest being stored in storage (column 7 lines 44-50, column 13 lines 10-59);

a signature generator coupled to the manifest generator coupled to the manifest generator to generate a manifest signature using a private key, the private key being decrypted by a decryptor using the OSNK, the manifest signature being stored in the storage (column 7 lines 44-50, column 13 lines 10-59);

a signature verifier to verify the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

a manifest verifier to verify the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the test center generating a pass/fail signal to

indicate whether the subset has been modified (column 7 lines 44-50, column 13 lines 10-59).

A manifest is described as a "descriptor" or as "representing the subset in a concise manner." England discloses a representation of a component of code in an OS, "the identity is a cryptographic digest of the code for the component, or a well-known name, or any other sting that is uniquely associated with the component." This can be interpreted as a "manifest" and the system of producing it as a "manifest generator." England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." Also, England discloses "appending the identity of each loaded component" and "signing the boot log to attest to its validity." The signing of the boot log represents a signature generator that is present, and a verifier to verify the validity of the signed component. Also, the manifest verifier is encompassed in the verification that the "boot log has not been tampered with" by comparing the cryptographic digests of the manifest created for each of the components.

Claim 19 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein protecting usage comprises:

decrypting a protected private key to generate a private key using the OSNK (column 7 lines 44-50, column 13 lines 10-59);

generating a signature of the subset of the software environment using the private key, the signature being stored in a storage (column 7 lines 44-50, column 13 lines 10-59); and

verifying the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/ not modified flag indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." It is obvious that the capability exists in the apparatus of England to decrypt a protected private key. Also, it is described that the processor has the capability to generate signatures, and the verification procedure for a signature is analogous to the comparator of the one-way hash functions.

Claim 20 is rejected as applied above in rejecting claim 13. Furthermore, England discloses:

The method of claim 13 wherein detecting comprises:

generating a manifest of the subset of the software environment, the manifest describing the subset of the software environment, the manifest being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

generating a manifest signature of the manifest using a private key, the private key being decrypted using the OSNK, the manifest signature being stored in the storage (column 7 lines 44-50, column 13 lines 10-59);

verifying the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

verifying the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the test center generating a pass/fail signal, the pass/fail signal indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

A manifest is described as a "descriptor" or as "representing the subset in a concise manner." England discloses a representation of a component of code in an OS, "the identity is a cryptographic digest of the code for the component, or a well-known name, or any other sting that is uniquely associated with the component." This can be interpreted as a "manifest" and the system of producing it as a "manifest generator." England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." Also, England discloses "appending the identity of each loaded component" and "signing the boot log to attest to its validity." The signing of the boot log represents a signature generator that is present, and a verifier to verify the validity of the signed component. Also, the manifest verifier is

encompassed in the verification that the "boot log has not been tampered with" by comparing the cryptographic digests of the manifest created for each of the components.

Claim 31 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

The computer program product of claim 25 wherein the computer readable program code for protecting usage comprises:

computer readable program code for decrypting a protected private key to generate a private key using the OSNK (column 7 lines 44-50, column 13 lines 10-59);

computer readable program code for generating a signature of the subset of the software environment using the private key, the signature being stored in a storage (column 7 lines 44-50, column 13 lines 10-59); and

computer readable program code for verifying the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." It is obvious that the capability exists in the apparatus of England to decrypt a protected private key. Also, it is described that the processor has the capability to generate signatures, and the

verification procedure for a signature is analogous to the comparator of the one-way hash functions.

Claim 32 is rejected as applied above in rejecting claim 25. Furthermore, England discloses:

The computer program product of claim 25 wherein the computer readable program code for protecting usage comprises:

computer readable program code for generating a manifest of the subset of the software environment, the manifest being stored in a storage (column 7 lines 44-50, column 13 lines 10-59);

computer readable program code for generating a manifest signature of the manifest using a private key, the private key being decrypted using the OSNK, the manifest signature being stored in the storage (column 7 lines 44-50, column 13 lines 10-59);

computer readable program code for verifying the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

computer readable program code for verifying the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the test center generating a pass/fail signal, the pass/fail signal indicating whether the subset of the software environment has been modified (column 7 lines 44-50, column 13 lines 10-59).

A manifest is described as a "descriptor" or as "representing the subset in a concise manner." England discloses a representation of a component of code in an OS, "the identity is a cryptographic digest of the code for the component, or a well-known name, or any other sting that is uniquely associated with the component." This can be interpreted as a "manifest" and the system of producing it as a "manifest generator." England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." Also, England discloses "appending the identity of each loaded component" and "signing the boot log to attest to its validity." The signing of the boot log represents a signature generator that is present, and a verifier to verify the validity of the signed component. Also, the manifest verifier is encompassed in the verification that the "boot log has not been tampered with" by comparing the cryptographic digests of the manifest created for each of the components.

Claim 43 is rejected as applied above in rejecting claim 37. Furthermore, England discloses:

The system of claim 37 wherein the usage protector comprises:

a decryptor to decrypt a protected private key to generate a private key using the OSNK (column 7 lines 44-50, column 13 lines 10-59);

a signature generator coupled to the decryptor to generate a signature of the subset of the software environment using the private key, the signature being stored in a storage (column 7 lines 44-50, column 13 lines 10-59); and

a signature verifier to verify the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the subset has been modified (column 7 lines 44-50, column 13 lines 10-59).

England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." It is obvious that the capability exists in the apparatus of England to decrypt a protected private key. Also, it is described that the processor has the capability to generate signatures, and the verification procedure for a signature is analogous to the comparator of the one-way hash functions.

Claim 44 is rejected as applied above in rejecting claim 37. Furthermore England discloses:

The system of claim 37 wherein the usage protector comprises:

a manifest generator to generate a manifest of the subset of the software environment, the manifest describing the subset of the software environment, the manifest being stored in storage (column 7 lines 44-50, column 13 lines 10-59);

a signature generator coupled to the manifest generator coupled to the manifest generator to generate a manifest signature using a private key, the private key being decrypted by a decryptor using the OSNK, the manifest signature being stored in the storage (column 7 lines 44-50, column 13 lines 10-59);

a signature verifier to verify the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage (column 7 lines 44-50, column 13 lines 10-59); and

a manifest verifier to verify the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the test center generating a pass/fail signal to indicate whether the subset has been modified (column 7 lines 44-50, column 13 lines 10-59).

A manifest is described as a "descriptor" or as "representing the subset in a concise manner." England discloses a representation of a component of code in an OS, "the identity is a cryptographic digest of the code for the component, or a well-known name, or any other sting that is uniquely associated with the component." This can be interpreted as a "manifest" and the system of producing it as a "manifest generator." England discloses that a "CPU 140 is capable of performing cryptographic functions, such as signing, encrypting, decrypting, and authenticating." Also, England discloses "appending the identity of each loaded component" and "signing the boot log to attest to its validity." The signing of the boot log represents a signature generator that is present,

and a verifier to verify the validity of the signed component. Also, the manifest verifier is encompassed in the verification that the "boot log has not been tampered with" by comparing the cryptographic digests of the manifest created for each of the components.

Conclusion

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kaveh Abrishamkar whose telephone number is 571-272-3786. The examiner can normally be reached on Monday thru Friday 8-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz Sheikh can be reached on 571-272-3795. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

KA
06/20/2005

AYAZ SHEIKH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100